# LearnJavaWeb

## Jsp

### JSP中的内置对象

在JSP中，Application= Servletcontext。

## MVC设计模式

浏览器将请求发送给servlet，servlet将请求中的数据封装在Javabean（request域）中，然后转发给jsp页面，最后返回jsp，显示给浏览器。

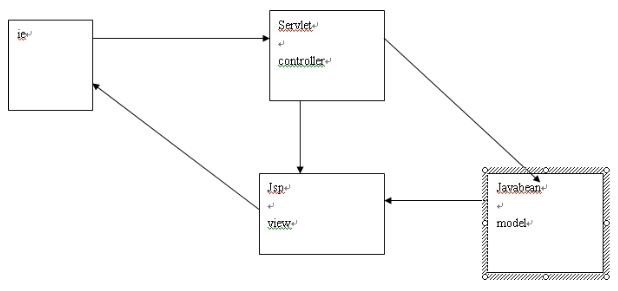


图 1 Jsp的MVC图示

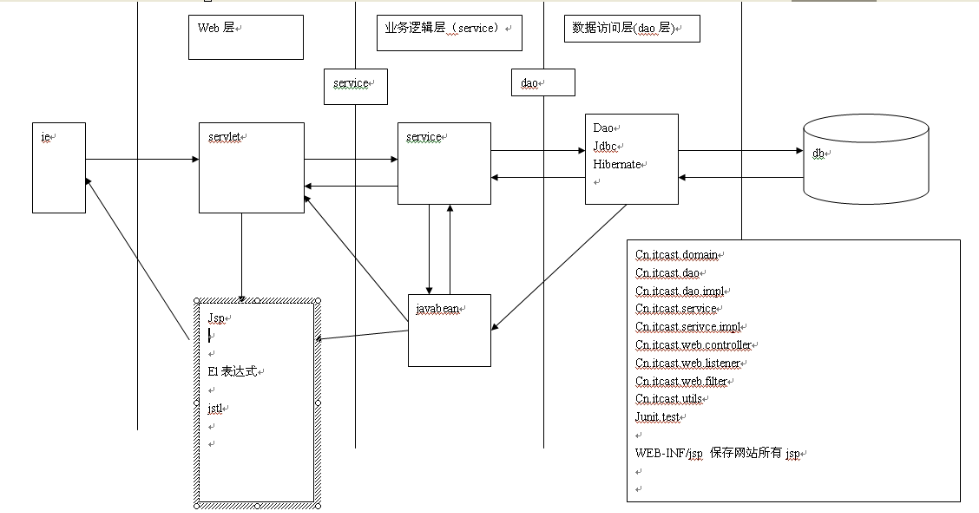


图 2软件开发层次示意图

## Strust

### Struct2项目配置步骤

### 自动注入

理解1：

先看一段代码  
假设你编写了两个类，一个是人(Person)，一个是手机(Mobile)。   
人有时候需要用手机打电话，需要用到手机的dialUp方法。   
传统的写法是这样:   
  
Java code   
  
public class Person{  
 public boolean makeCall(long number){  
 Mobile mobile=new Mobile();  
 return mobile.dialUp(number);  
 }  
}  
  
也就是说，类Person的makeCall方法对Mobile类具有依赖，必须手动生成一个新的实例new Mobile()才可以进行之后的工作。   
[依赖注入](https://www.baidu.com/s?wd=%E4%BE%9D%E8%B5%96%E6%B3%A8%E5%85%A5&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWm1P9nhNBuynvnHRsPHF90ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1b4rjTYn1mv)的思想是这样，当一个类(Person)对另一个类(Mobile)有依赖时，不再该类(Person)内部对依赖的类(Moblile)进行实例化，而是之前配置一个beans.xml,告诉容器所依赖的类(Mobile)，在实例化该类(Person)时，容器自动注入一个所依赖的类(Mobile)的实例。   
接口：   
  
Java code   
  
public Interface MobileInterface{  
 public boolean dialUp(long number);  
}  
  
Person类：   
  
Java code   
  
public class Person{  
 private MobileInterface mobileInterface;  
 public boolean makeCall(long number){  
 return this.mobileInterface.dialUp(number);  
 }  
 public void setMobileInterface(MobileInterface mobileInterface){  
 this.mobileInterface=mobileInterface;  
 }  
}  
在[xml文件](https://www.baidu.com/s?wd=xml%E6%96%87%E4%BB%B6&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWm1P9nhNBuynvnHRsPHF90ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3En1b4rjTYn1mv" \t "_blank)中配置依赖关系   
  
Java code   
<bean id="person" class="Person">  
 <property name="mobileInterface">  
 <ref local="mobileInterface"/>  
 </property>   
</bean>  
<bean id="mobileInterface" class="Mobile"/>  
这样，Person类在实现拨打电话的时候，并不知道Mobile类的存在，它只知道调用一个接口MobileInterface，而MobileInterface的具体实现是通过Mobile类完成，并在使用时由容器自动注入，这样大大降低了不同类间相互依赖的关系。

理解2：

IoC 或者 DI 或者 ...一大堆的缩写词  
不管是[面向对象](https://www.baidu.com/s?wd=%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnyRLnvF9uh7BuWcLmyNb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHnzrHD1nWR)，还是[面向过程](https://www.baidu.com/s?wd=%E9%9D%A2%E5%90%91%E8%BF%87%E7%A8%8B&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnyRLnvF9uh7BuWcLmyNb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHnzrHD1nWR)，都需要分成许多的块，然后由这些部件[协同工作](https://www.baidu.com/s?wd=%E5%8D%8F%E5%90%8C%E5%B7%A5%E4%BD%9C&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnyRLnvF9uh7BuWcLmyNb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHnzrHD1nWR)完成任务   
要[协同工作](https://www.baidu.com/s?wd=%E5%8D%8F%E5%90%8C%E5%B7%A5%E4%BD%9C&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnyRLnvF9uh7BuWcLmyNb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHnzrHD1nWR" \t "_blank)就会产生依赖，一个方法调用另一个方法，一个对象包含另一个对象   
如果对象A包含对象B的话，就需要在A里new一个B   
[依赖注入](https://www.baidu.com/s?wd=%E4%BE%9D%E8%B5%96%E6%B3%A8%E5%85%A5&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWnyRLnvF9uh7BuWcLmyNb0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHnzrHD1nWR)从具体类B里抽象出接口IB——IB的具体实现可能有很多B,B1,B2...很多种——这样A可以不用再new具体的B了，而是跟IoC容器说：我要一个IB（getBean("IB")）。然后，由容器根据配置文件来做具体的new的工作。具体new的是哪个，由配置文件从代码外部决定，要更换成B,B1,或是B2...修改配置文件就能做到，不用再改代码了

### 页面表单请求参数的注入到Action中的原理

在Struct2中，表单提交的数据（username）会自动注入到Action类中对应的属性（String username），前提是Action对象为这个属性提供了Setter（）方法进行注入。

### 注意事项

* 使用HTML的标签时，<form action=”userAction.action” />，需要.action。而使用struct2标签时，不需要<s:form action=”userAction” />

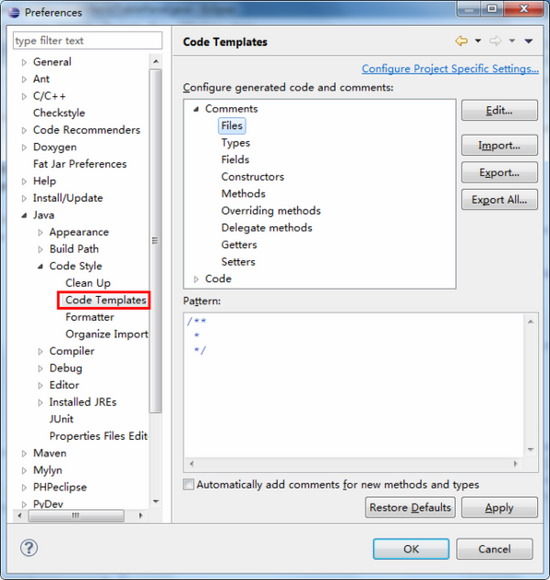
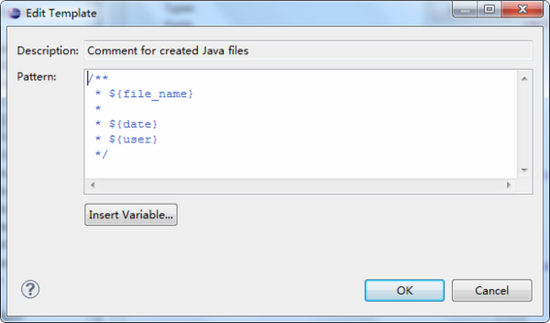
## Eclipse相关

### 函数添加注释说明

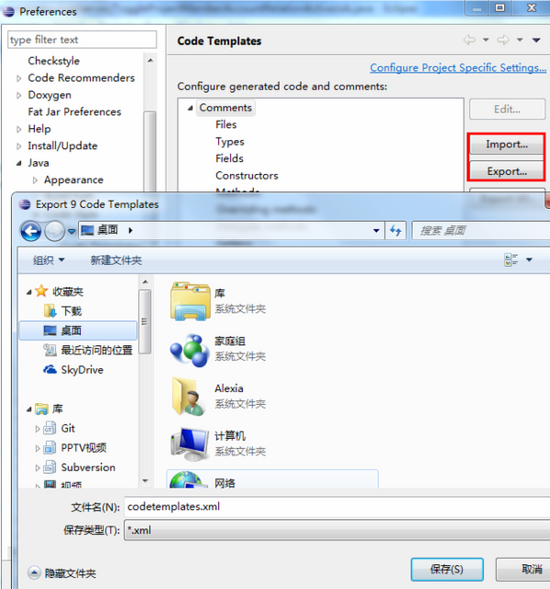
#### 快速注释说明

要把鼠标焦点放在方法前，然后ALT+SHIFT+J。

#### 自定义注释模板

1. Window->Preference 打开参数设置面板，然后选择：Java -> Code Style -> CodeTemplates
2. 可以看到Comments下面一系列可以注释的Java代码组成部分（对文件、变量、方法、构造方法、重载函数等等），比如我想定制Java文件的注释风格，那么在右侧选择Comments中的File项，点击Edit，进入编辑模式，这样就可以自定义注释了。其中可以插入一些变量，如日期、文件名、作者等等。

#### 导入/导出代码格式模板

如果你辛辛苦苦定制好了自己的代码风格，然后换了台机器进行操作或重装了Eclipse，是不是要重新配置一遍呢？答案当然是No了，Eclipse提供了“导出”和“导入”功能，你可以把自己的模板导出来在其他机器上使用。

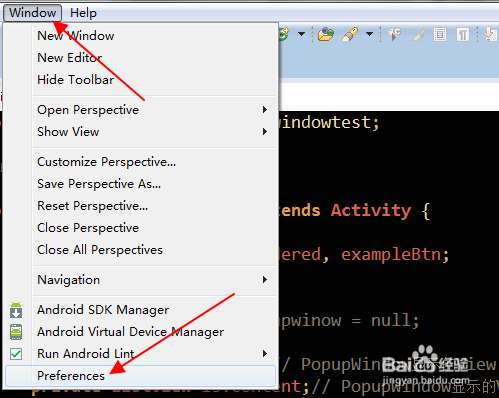
### 注释

|  |  |
| --- | --- |
| // 风格 | Ctrl+/ |
| /\*\*/ 风格 | Shift+Ctrl+/ |

### 乱码问题

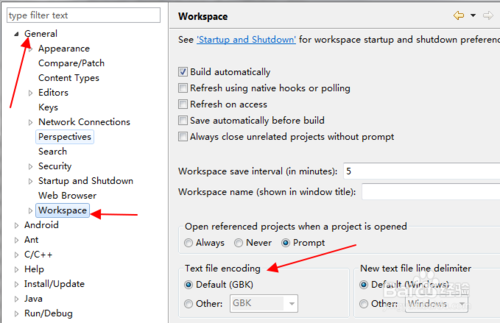
#### Eclipse默认编码

1. 编辑器的编码会影响到所有的项目中的字符的显示，可以说是作用最为广泛的设置，每一个项目都会受到这个设置的影响。点击菜单栏中的窗口（Window）— —选项（Preferences）。

[](http://jingyan.baidu.com/album/870c6fc325a691b03fe4beac.html?picindex=2)

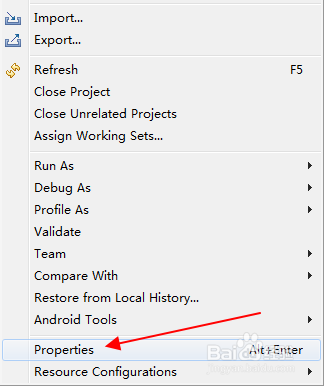
1. 2

点击常规（General）— —工作空间（Wordspace），然后在右侧的文本文件编码格式（Text file encoding）中选择你想要设置的编码格式，系统默认的格式为GBK，建议改为UTF-8

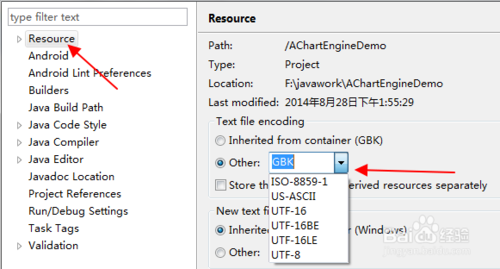
[](http://jingyan.baidu.com/album/870c6fc325a691b03fe4beac.html?picindex=3)

#### 整个项目编码

1. 如果其他的项目中不存在乱码问题，只是个别项目显示时出现乱码，那么我们不需要设置工作空间的编码，只需要修改项目的编码即可。
2. 选中项目后，右键点击，在弹出的菜单中点击属性（Properties）。

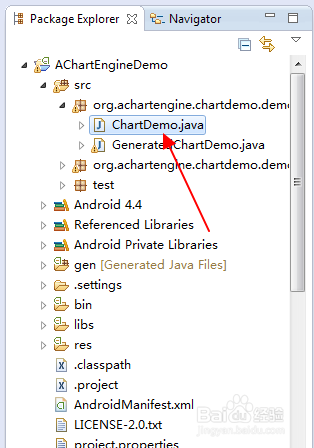
[](http://jingyan.baidu.com/album/870c6fc325a691b03fe4beac.html?picindex=4)

1. 3在弹出的属性窗口中，选择资源（Resource）— —点击选择Other（其他），然后选择你想要的编码格式。

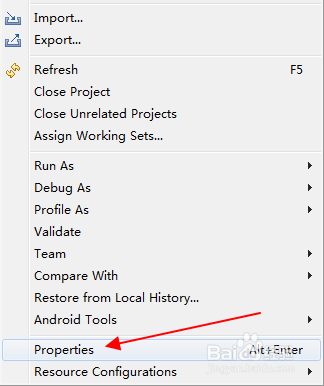
[](http://jingyan.baidu.com/album/870c6fc325a691b03fe4beac.html?picindex=5)

#### 单个文件编码

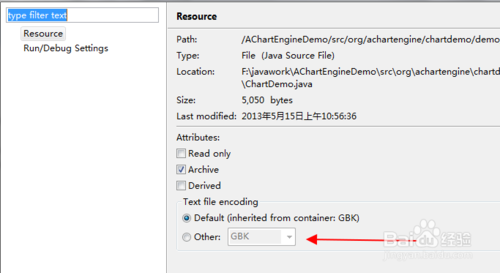
1. 只是个别文件出现了乱码，那么设置个别文件的编码格式就可以解决问题了。选中有乱码的文件，然后点击右键。

[](http://jingyan.baidu.com/album/870c6fc325a691b03fe4beac.html?picindex=6)

1. 在弹出的菜单中选择属性（Properties）。

[](http://jingyan.baidu.com/album/870c6fc325a691b03fe4beac.html?picindex=7)

3在弹出的属性窗口中选择资源，点击Other（其他）后设置个别文件的编码方式。

[](http://jingyan.baidu.com/album/870c6fc325a691b03fe4beac.html?picindex=8)